

1503.65719

PATENT APPLICATION

#3  
11002 U.S. PTO  
09/911784  
07/24/01

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re U.S. Patent Application )  
Applicant: Yoshiyama et al. )  
Serial No. )  
Filed: July 24, 2001 )  
For: DATABASE RETRIEVING )  
METHOD, APPARATUS )  
AND STORAGE MEDIUM )  
THEREOF )  
Art Unit: )

*I hereby certify that this paper is being deposited with the United States Postal Service as EXPRESS MAIL in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231, on July 24, 2001*  
*Express Label No.: EL846221708US*  
Signature: David P. Crain  
EXPRESS.WCM  
Appr. February 20, 1998

CLAIM FOR PRIORITY

Assistant Commissioner for Patents  
Washington, DC 20231

Sir:

Applicant claims foreign priority benefits under 35 U.S.C. § 119 on the basis of the foreign application identified below:

Japanese Patent Application No. 2001-053884, filed February 28, 2001.

A certified copy of the priority document is enclosed.

Respectfully submitted,

GREER, BURNS & CRAIN, LTD.

By Patrick G. Burns  
Reg. No. 29,367

July 24, 2001  
300 South Wacker Drive  
Suite 2500  
Chicago, IL 60606  
(312) 360-0080  
Customer Number: 24978

PATENT OFFICE  
JAPANESE GOVERNMENT

This is to certify that the annexed is a true copy of the following application as filed with this Office.

Date of Application: February 28, 2001

Application Number: Patent Application  
No. 2001-053884

Applicant(s): FUJITSU LIMITED

May 31, 2001

Commissioner,  
Patent Office Kozo Oikawa

Certificate No. 2001-3049188

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

11002 U.S. PRO  
09/911784  
07/24/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されて  
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed  
with this Office

出 願 年 月 日

Date of Application:

2001年 2月28日

出 願 番 号

Application Number:

特願2001-053884

出 願 人

Applicant(s):

富士通株式会社

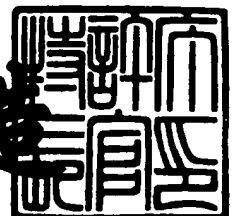
CERTIFIED COPY OF  
PRIORITY DOCUMENT

BEST AVAILABLE COPY

2001年 5月31日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3049188

【書類名】 特許願

【整理番号】 0052439

【提出日】 平成13年 2月28日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 17/30

【発明の名称】 データベース検索方法ならびに記憶媒体

【請求項の数】 5

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 吉山 正治

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 脇山 克士

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100074099

【住所又は居所】 東京都千代田区二番町8番地20 二番町ビル3F

【弁理士】

【氏名又は名称】 大菅 義之

【電話番号】 03-3238-0031

【選任した代理人】

【識別番号】 100067987

【住所又は居所】 神奈川県横浜市鶴見区北寺尾7-25-28-503

【弁理士】

【氏名又は名称】 久木元 彰

【電話番号】 045-573-3683

【手数料の表示】

【予納台帳番号】 012542

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9705047

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データベース検索方法ならびに記憶媒体

【特許請求の範囲】

【請求項 1】 データベースの検索を行う計算機によって使用されるプログラムにおいて、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手順と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、該検索条件に対応するインデックスを作成する手順と、

該作成されたインデックスを用いてデータベースの検索を行う手順とを計算機に実行させるためのプログラム。

【請求項 2】 データベースの検索を行う計算機によって使用されるプログラムにおいて、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手順と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みのインデックスのうちで、前記検索条件よりも広い条件を満足するインデックスが存在するか否かを判定する手順と、

該広い条件を満足するインデックスが存在する時、該インデックスを用いて該検索条件のみを満足するインデックスを作成する手順と、

該作成されたインデックスを用いてデータベースの検索を行う手順とを計算機に実行させるためのプログラム。

【請求項 3】 データベースの検索を行う計算機によって使用されるプログラムにおいて、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手順と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みの複数のインデックスのうちで、組合わせることによって前記検索条件を満足する 2 つ以上のインデックスが存在するか否かを判定する手順と、

該 2 つ以上のインデックスが存在する時、該 2 つ以上のインデックスを組合わせて、該検索条件に対応するインデックスを作成する手順と、

該作成されたインデックスを用いてデータベースの検索を行う手順とを計算機に実行させるためのプログラム。

【請求項 4】 前記プログラムが、

前記作成されたインデックスに対するアクセス回数、作成日時、更新頻度のデータを管理する手順と、

該データの管理状況に対して、作成されたインデックスの消去を行う手順とを更に備えることを特徴とする請求項 1、2、または 3 記載のプログラム。

【請求項 5】 前記プログラムが、

前記データベースへのアクセスがデータの更新、または削除のためのアクセスである時、該アクセス処理に適用可能な作成済みインデックスが存在するか否かを判定する手順と、

該インデックスが存在する時、該インデックスの存在によって該アクセス処理のアクセス性能が低下するか否かを判定する手順と、

該アクセス性能が低下する時、該アクセス処理の開始に先立って、該インデックスを削除する手順とを更に備えることを特徴とする請求項 1、2、3 または 4 記載のプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明はデータベース、特に関係データベースの検索方式に係り、更に詳しくは特定の条件での検索処理、すなわち不定形の検索処理を行う場合に、その特定の検索処理に適合するインデックスを動的に作成して、データベースの検索を行うデータベース検索方法、およびその方法を格納した記憶媒体に関する。

【0002】

【従来の技術と発明が解決しようとする課題】

近年、関係データベースに対する検索方式としての、頻繁に実行される業務に対応する定形の検索だけではなく、データウェアハウスのようにその度に検索条

件が異なる不定形の検索処理に対しても高速性が要求されている。

【 0 0 0 3 】

データベースに対してユーザが行う操作としてはデータの登録、検索、更新、および削除の4種類がある。これらはまとめてデータ操作と呼ばれ、データ操作を実行するシステムとしてDBMS（データベースマネジメントシステム）が使用される。

【 0 0 0 4 】

DBMSによってデータ操作を行うための言語はDML（データマニピュレーションランゲージ、データ操作言語）と呼ばれ、その1つの種類としてSQL（ストラクチャード クエリー ランゲージ、構造化問い合わせ言語）がある。

【 0 0 0 5 】

従来SQL言語を用いてデータベースの検索を行う場合には、検索に必要なインデックスを事前に作成した後に検索が行われていた。定形の検索では、作成されたインデックスを検索のたびに利用することによってデータベースの検索を高速に行うことができた。

【 0 0 0 6 】

これに対してその度に検索条件が異なる不定形の検索を行う場合には、すでに作成済みのインデックスを使用できない場合が多く、そのため不定形の検索を高速に行うためには検索に必要なインデックスをあらかじめ予想して、あらゆる検索条件に対応するインデックスを作成しておくか、またはインデックスを使用せずに全件走査による検索を行っていた。またインデックスを作成しておく場合には、データベースのデータの更新や削除を行うたびに、そのデータに対応するインデックスの更新を行っていた。

【 0 0 0 7 】

このようにSQL言語を用いたデータベースの検索において、検索に必要な適当なインデックスが存在しない場合には、データベースへのアクセスは全件走査となる。全件走査ではインデックスを使用したアクセスと比較して、大量の時間とCPU資源を消費してしまうことになる。

【 0 0 0 8 】



この問題点を避けて検索を高速化するためには、検索条件に一致するインデックスをあらかじめ作成しておく必要があり、不定形の検索が行われる場合には特定の検索条件に一致するインデックスをそれぞれの条件毎に作成しておく必要があるために、インデックスを格納するためのディスク資源が大量に消費されてしまうという問題点があった。またインデックスを多く作成することによって、データの更新／追加／削除時のアクセス性能の低下を招いてしまうという問題点もあった。

## 【 0 0 0 9 】

本発明の課題は、上述の問題点に鑑み、データベースの検索時に必要となるインデックスを動的に作成したり、また作成されているインデックスを再利用することによって、インデックスが存在しないか、または存在してもそのままでは使用できない場合のデータベースの検索を高速化することと、作成されたインデックスを格納するための資源を削減することと、データベースのデータ検索時のアクセス性能の劣化を防止することである。

## 【 0 0 1 0 】

## 【課題を解決するための手段】

図 1 は本発明の機能ブロック図である。同図はデータベースの検索を行う計算機によって使用される記憶媒体、例えばフロッピーディスクなどの計算機読出し可能可搬型記憶媒体に格納され、本発明のデータベース検索方式を実現するためのプログラムの機能ブロック図である。

## 【 0 0 1 1 】

このプログラムでは、図 1 の 1 で検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較するステップが、2 でその比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みのインデックスのうちで検索条件を満足し、適用可能なインデックスが存在するか否かを判定するステップが、3 で適用可能なインデックスが存在しない時、検索条件に対応するインデックスを作成するステップが、そして 4 で作成されたインデックスを用いてデータベースの検索を行うステップが、計算機によって実行される。

## 【 0 0 1 2 】

本発明においては、全件検索を行う場合の方がコストが大きい時、図 1 の 2 で適用可能なインデックスの存在の有無を判定することなく、直ちに 3 のステップ、すなわち検索条件に対応するインデックスを作成し、そのインデックスを用いて検索を行うこともできる。

## 【 0 0 1 3 】

また本発明の記憶媒体に格納されるプログラムとしては、検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較するステップと、そのコスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みのインデックスのうちで検索条件よりも広い条件を満足するインデックスが存在するか否かを判定するステップと、そのような広い条件を満足するインデックスが存在する時、そのインデックスを用いて検索条件のみを満足するインデックスを作成するステップと、作成されたインデックスを用いてデータベースの検索を行うステップとを計算機に実行させるためのプログラムを用いることもできる。

## 【 0 0 1 4 】

更に本発明の記憶媒体に格納されるプログラムとして、検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較するステップと、そのコスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みの複数のインデックスのうちで、組合わせることによって検索条件を満足する 2 つ以上のインデックスが存在するか否かを判定するステップと、そのような 2 つ以上のインデックスが存在する時、2 つ以上のインデックスを組合わせて検索条件を満足するインデックスを作成するステップと、作成されたインデックスを用いてデータベースの検索を行うステップとを計算機に実行させるためのプログラムを用いることもできる。

## 【 0 0 1 5 】

また本発明においては、以上のような記憶媒体に格納されたプログラムが、それぞれ、作成されたインデックスに対するアクセス回数、作成日時、更新頻度のデータを管理するステップと、その管理状況に対応して作成されたインデックス

の消去を行うステップとを更に備えることもできる。

【 0 0 1 6 】

発明の実施の形態においては、以上のようなプログラムが、データベースへのアクセスがデータの更新、または削除のためのアクセスである時、そのアクセス処理に適用可能な作成済みインデックスが存在するか否かを判定するステップと、そのようなインデックスが存在する時、そのインデックスの存在によってアクセス処理のアクセス性能が低下するか否かを判定するステップと、アクセス性能が低下する時、そのアクセス処理に先立ってそのインデックスを削除するステップとを更に備えることもできる。

【 0 0 1 7 】

本発明においてデータベースの検索を行うデータベース検索方法においては、以上に述べた記憶媒体に格納されたプログラムと同様の手順を用いるデータベース検索方法が用いられる。

【 0 0 1 8 】

本発明においてデータベースの検索を行うデータベース検索装置は、検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較するコスト比較手段と、そのコスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みのインデックスのうちで検索条件を満足し、適用可能なインデックスが存在するか否かを判定する適用可能インデックス判定手段と、適用可能なインデックスが存在しない時、検索条件に対応するインデックスを作成するインデックス作成手段と、作成されたインデックスを用いてデータベースの検索を行う検索手段とを備える。

【 0 0 1 9 】

またこのデータベース検索装置は、作成されたインデックスのデータ作成条件、およびインデックス格納領域の管理を行うインデックス管理手段を更に備えることもできる。

【 0 0 2 0 】

以上のように本発明によれば、特に不定形のデータベース検索を行う場合に、必要となるインデックスを動的に作成するか、または作成済みのインデックスを

再利用することが可能となる。

【 0 0 2 1 】

【発明の実施の形態】

図 2 は本発明のデータベース検索方式が用いられるデータベース検索システムの基本構成ブロック図である。同図において、システムの中心的役割を果たすものはデータベース検索装置 1 0 である。

【 0 0 2 2 】

データベース検索装置 1 0 は、アプリケーションプログラム 1 1 からデータベース検索要求としての S Q L 文が与えられると、その S Q L 文の解析、およびデータベースの検索条件の解析を実行する構文解析部 1 2、構文解析部 1 2 の解析結果に従って最も速度が速くなるアクセス方法を決定するアクセス処理最適化部 1 3、アクセス処理最適化部 1 3 によって決定されたアクセス方法を用いてデータベースに対するアクセスを実行するアクセス処理部 1 4、データベースに対するインデックスを管理するインデックス管理部 1 5、データベース内で実際にデータを格納している表 1 6 に対する、一般に複数のインデックス 1 7 の中で、データベースへのアクセスに際して適用可能なインデックスが存在しない場合に動的インデックス 1 9 を作成したり、作成されている動的インデックスを消去するインデックス作成／消去部 1 8、これらの動的インデックスの寿命を管理する寿命管理部 2 0 から構成されている。

【 0 0 2 3 】

図 3 は本発明の実施形態における S Q L 処理システムの構成ブロック図である。同図の構成は、図 2 のデータベース検索システムの構成と類似しているが、データベース検索装置 1 0 に対応する S Q L 処理装置 3 0 の内部に管理辞書 4 1 が備えられている点が基本的に異なっている。

【 0 0 2 4 】

図 3 の S Q L 処理装置 3 0 の動作について更に説明すると、前述のように構文解析処理装置 3 2 によって S Q L 文および検索条件が解析され、アクセス処理最適化処理装置 3 3 によってその解析結果に応じて、例えば新たに動的インデックスを作成してアクセスを行う場合と、動的インデックスを作成せずにデータベー

スの全件検索を行う場合とでどちらが処理が速くなるかが比較され、その比較結果に応じてアクセス処理装置 3 4 によってデータベースに対するアクセスが実行される。

【 0 0 2 5 】

この時、動的インデックスを作成した方が処理が速いと判定されると、動的インデックス作成装置 3 8 によって動的インデックス 3 9 が作成され、その作成された動的インデックスを用いてデータベースの表 3 6 に対するアクセスが実行される。

【 0 0 2 6 】

また、アクセス処理最適化処理装置 3 3 はアクセス方法の決定にあたって、すでに作成されているインデックス 3 7、または動的インデックス 3 9 を使用することができないか否かについてインデックス管理装置 3 5 に問い合わせを行ない、インデックス 3 7 および動的インデックス 3 9 を管理しているインデックス管理装置 3 5 は、アクセスに使用できる適切なインデックスがある場合には、アクセス処理最適化処理装置 3 3 に対してそれを通知し、アクセス処理装置 3 4 によってそのインデックス、または動的インデックスを用いて、データベースの表 3 6 に対するアクセスを行うことができる。

【 0 0 2 7 】

動的インデックス作成装置 3 8 は、例えばデータベースへのアクセスがデータの更新である場合に、そのデータ更新と共に動的インデックスの更新も必要となり、その結果アクセス性能が低下するとアクセス処理最適化処理装置 3 3 によって判定された場合に、動的インデックス 3 9 の削除を実行する。

【 0 0 2 8 】

またインデックス管理装置 3 5 によって管理されているインデックス 3 7、および動的インデックス 3 9 の一部を使用してデータベースにアクセスを行うことができる場合や、これらのインデックスを複数組合わせてアクセスを行うことができる場合には、アクセス処理装置 3 4 はインデックスの一部、または組合わせによって作成される動的インデックス 3 9 を用いてデータベースに対するアクセスを実行することができる。

## 【 0 0 2 9 】

またインデックス 3 7 や動的インデックス 3 9 に対して、新たな条件を追加してインデックスを作成することによりアクセスができる場合は、そのようなインデックスが作成されて、アクセス処理装置 3 4 によるアクセスが行われる。

## 【 0 0 3 0 】

動的インデックス作成装置 3 8 によって作成された動的インデックスの情報、すなわち作成条件、サイズ、列の情報などはインデックス管理装置 3 5 に送られ、インデックス管理装置 3 5 はこれらの情報と通常のインデックス 3 7 の情報を管理辞書 4 1 に格納して管理する。管理辞書 4 1 の内容については後述する。

## 【 0 0 3 1 】

寿命管理装置 4 0 は、動的インデックス作成装置 3 8 によって、例えばメモリの作業領域に作成された動的インデックスの寿命、すなわち有効期間を管理するものである。この寿命はインデックスが作成されてからの経過時間や、アクセスの回数、および更新頻度などによって管理される。

## 【 0 0 3 2 】

これによって、例えば更新頻度の高いものから順に動的インデックス作成装置 3 8 に対して動的インデックスの削除要求が出され、動的インデックスの削除が行われる。またデータベースへのアクセスによって大量にデータの更新が行われる場合には、動的インデックスの更新も必要となるため、あらかじめ動的インデックスの削除が行われる。

## 【 0 0 3 3 】

図 4 はデータベースを構成する会員表の例である。この表は会員番号、年齢、および氏名の 3 つの項目（列）によって構成されており、1 行に 1 件、合計 9 9 9 9 件のデータ（レコード）が格納されているものとする。

## 【 0 0 3 4 】

図 5 は、データベースへのアクセスがデータの参照である場合のデータベースアクセス処理のフローチャートである。同図において処理が開始されると、まずステップ S 1 で構文解析処理が行われる。この処理では、図 3 の S Q L 処理装置 3 0 に対してアプリケーションプログラム 3 1 から入力された S Q L 文の構文解

析が行われ、SQL文の種類、アクセスすべき表、および検索条件などの情報が解析され、解析された情報は次のステップS2におけるアクセス処理最適化処理のための情報として記憶される。

## 【0035】

ステップS2のアクセス処理最適化処理は、前述のようにデータベースへのアクセスが最も速くなるアクセス方法を決定するものであり、ステップS3～S9の処理によって構成される。

## 【0036】

このアクセス処理最適化処理においては、例えば表が複数あってジョイン（結合）の処理が必要になる場合などは、それらの表へのアクセス順序と、アクセス手順、すなわち全件走査か、インデックスアクセスかなどが決定される。アクセス手順は、アクセスが最も速くなるように、データの総件数、データの分布、検索条件などから決定される。

## 【0037】

図4の表に対して次のようなSQL文が入力された場合を考える。

```
SELECT * FROM 会員表 WHERE 会員番号 BETWEEN 'A4500' AND 'A6500' ORDER BY 会員番号
```

ステップS1における構文解析処理の結果、検索条件の対象となる列は会員番号であり、その条件はA4500とA6500の間であり、更に列としての会員番号で昇順にソートを行う必要があることになる。

## 【0038】

次にアクセス処理最適化処理、すなわちステップS2の内部のステップS3のコスト計算の処理として、列としての会員番号に対して検索条件に対応する動的インデックスを作成してアクセスを行った場合と、すでに存在するインデックス、または動的インデックスを用いてアクセスを行った場合と、9999件のデータの全件走査を行った場合とのコスト計算が行われ、ステップS4で全件走査した方が速いか否かが判定される。

## 【0039】

ステップS4で全件走査を行う場合よりも、例えば動的インデックスを作成し

てからアクセスを行った方が速いと判定されると、ステップ S 5 ですでにインデックスが存在するか、すなわち図 3 のインデックス 3 7 が存在するか否かが判定され、存在する場合にはステップ S 6 でそのインデックスが適用可能か否かが判定される。そのインデックスが適用可能な場合には、そのインデックスを使用してステップ S 7 でアクセス処理が実行される。

## 【 0 0 4 0 】

これに対してステップ S 5 でインデックスが存在しないか、あるいは存在してもステップ S 6 で存在するインデックスが適用不可能と判定されると、ステップ S 8 ですでに動的インデックスが存在するか、すなわち図 3 の動的インデックス 3 9 が作成されているか否かが判定される。すでに存在する場合には、ステップ S 9 でその動的インデックスが適用可能か否かが判定され、適用可能な場合にはステップ S 7 でアクセス処理が実行される。

## 【 0 0 4 1 】

ステップ S 8 で動的インデックスが存在しないか、存在してもステップ S 9 でその動的インデックスが適用不可能と判定されると、ステップ S 1 0 で動的インデックスが作成され、その動的インデックスを用いてステップ S 7 でアクセス処理が実行される。またステップ S 4 で、例えば動的インデックスを作成してからアクセスを行うよりも全件走査した方が速いと判定されると、直ちにステップ S 7 で全件走査によってアクセス処理が実行される。

## 【 0 0 4 2 】

なおステップ S 4 で、例えば動的インデックスを新たに作成してアクセスを行った場合の方が全件走査を行う場合よりも速いと判定されても、すでに作成されているインデックス、または動的インデックスを利用する方が更にアクセス処理は高速になると考えられるので、ステップ S 5 ～ S 9 の処理が行われ、すでに作成されているインデックス、または動的インデックスを適用できる場合には、そのインデックスを適用してアクセス処理が実行される。

## 【 0 0 4 3 】

ここでインデックスを使用してアクセスを行う場合と、インデックスを使用せずに全件走査を行う場合とのアクセスの速さについて説明する。話を単純にす



るために、1件だけしかレコード（データ）がない場合を考えると、インデックスを利用してデータを読む場合には、まず第1にインデックスの読みが行われ、次に例えばアドレスの値が検索され、更にそのアドレスの値に対するファイルの読みが行われるため、入／出力が2回必要となる。インデックスを使用しない場合にはファイルの読みだけですみ、すなわち入／出力は1回となる。このため1件のデータの読みではインデックスを使用した方が遅くなる。

## 【 0 0 4 4 】

インデックスを利用したアクセスが高速となるかは、全体のデータ件数のうちで何件を読み込むかによる。例えば10万件のデータのうち条件に該当する1件だけを読むのであれば、当然インデックスを利用した方が高速になるが、10万件のうち条件に該当する9万件を読み込む場合には全件走査を行う方が高速になる。

## 【 0 0 4 5 】

すなわち、インデックスがあるからといって常にインデックスアクセスを行うというような単純な処理を行うことはできず、コスト計算によってどのアクセス手順が最適になるかを判断する必要がある。特に条件が複数ある場合などは、それぞれの条件に対してインデックスを使用してアクセスした場合と、全件走査によってアクセスした場合をあらかじめ比較する必要がある。

## 【 0 0 4 6 】

図6はデータベースへのアクセスがデータの更新や削除である場合のアクセス処理のフローチャートである。同図において処理が開始されると、まずステップS11で、図5のステップS1におけると同様に、構文解析処理が行われる。この処理では、前述と同様にアプリケーションプログラムから入力されたSQL文の種類、アクセスすべき表、検索条件などの情報が解析され、解析された情報はSQL処理装置30の内部のテーブルに記憶される。

## 【 0 0 4 7 】

ここで図5のステップS10によって作成された動的インデックスとして、前述のSQL文に対応する会員番号がA4500以上、A6500以下の条件に対する動的インデックスがすでに存在するものとし、更にデータベース内のデータ

ベースの削除のために次のようなSQL文が入力された場合を考える。

【0048】

DELETE FROM 会員表 WHERE 会員番号>'A0500'

このSQL文の入力に対応して、図6のステップS12のアクセス処理最適化処理では、まずステップS13で適用可能なインデックスがあるか否かが判定され、ここでは会員番号がA0500以上に相当するインデックスが存在しないために、ステップS14で検索条件に含まれるインデックスがあるか否かが判定される。

【0049】

会員番号がA4500以上、A6500以下の動的インデックスは検索条件の中に含まれているため、この動的インデックスに対応するデータは全て削除されることになってこの動的インデックス自体が無効となり、ステップS15でインデックスの削除処理が行われた後に、ステップS16で実際にデータベースに対するアクセス処理、すなわちデータの削除処理が実行される。

【0050】

これに対して前述の動的インデックスが存在し、更に次のようなSQL文が入力された場合を考える。

DELETE FROM 会員表 WHERE 会員番号 ≥ 'A5000'  
AND 会員番号 ≤ 'A5010'

この削除処理では会員番号がA5000以上、A5010以下のデータの削除が行われることになるが、ステップS13で適用可能なインデックスが存在すると判定され、ステップS14でコスト計算が行われる。この場合削除されるデータの件数自体が少なく、また動的インデックスの対応する部分の削除の処理にも時間がかからないため、ステップS15で動的インデックスの存在により性能が劣化するか否かの判定において劣化しないと判定され、動的インデックスの削除を行うことなく、ステップS16のアクセス処理が実行される。

【0051】

これに対して、例えば動的インデックスに対応する会員番号のデータのうちに削除されるデータの件数が多く、動的インデックスの存在によって性能が劣化す

ると判定される場合には、ステップ S 1 5 でその動的インデックスの削除処理が行われた後に、ステップ S 1 6 でアクセス処理が行われる。更にステップ S 1 4 で条件に含まれるインデックスがないと判定された場合には、直ちにステップ S 1 6 でアクセス処理が実行される。このように本実施形態ではデータ（レコード）の削除や更新に際しても動的インデックスの存在による性能劣化が防止できる。

#### 【 0 0 5 2 】

ここでデータベースのデータの削除のコストについて説明する。データの参照の場合と同様に、レコード（データ）を 1 件だけ削除するとき、インデックスを使用する場合にはまずインデックスを参照し、例えばアドレスの値を判定して、実際にデータを削除することになる。インデックスを使用しない場合には、アドレスの値を判定して実際のデータを削除すればよいことになり、インデックスを使用する場合の方がインデックスの参照が入るために遅くなる。

#### 【 0 0 5 3 】

但し、これは 1 件のレコードだけを削除する場合であり、例えば 1 0 万件のデータのうち 1 件のデータを削除する場合には、インデックスを使用の方が全件検索を行うより処理は速くなる。インデックスを使う方が速いか否かはデータベースの特性に依存するため、何%以上のデータを削除する場合に全件検索の方が速くなるかについては単純に判定できず、例えば全体の 5 0 % のデータが削除の対象となる場合にはインデックスを使用しないことにするなど、その判定の方法はまちまちである。

#### 【 0 0 5 4 】

図 7 ～ 図 9 は動的インデックス作成例の説明図である。図 7 は動的インデックスの作成対象となる列に対してインデックスが作成されていない場合の処理の例を示す。ここでは次の S Q L 文に対応して動的インデックスが作成されるものとする。

#### 【 0 0 5 5 】

```
SELECT * FROM 会員表 WHERE 年齢 ≥ 2 5 AND 年齢 ≤ 3 0
```

図3の動的インデックス作成装置38は、対象となる表のデータに対して全件走査を行い、一般的には最終レコードに至るまで、条件判断と探索条件判断とを繰返し、これらの条件を満足するデータを選択して動的インデックスを作成する。ここで条件と探索条件とを合わせたものがデータベースの検索条件に相当する。

#### 【0056】

図7では、条件判断において年齢が25歳以上、30歳以下のデータの判定を行うのみであり、探索条件判断は行われぬ。例えば図9で説明するように、条件判断によって年齢の条件を満足するデータに対して、更に会員番号の判断を行うような場合が探索条件判断に相当する。なお図7で作成された動的インデックスで、例えば第1行の"30"は会員番号A0030に対応し、会員表内のこのレコードの物理アドレスを示す。

#### 【0057】

動的インデックスは、例えばメモリの作業領域に作成され、作成された動的インデックスの作成条件、および表との関連の情報などは図3のインデックス管理装置35に送られる。インデックスの作成される作業領域はディスク上でもかまわない。

#### 【0058】

図8は動的インデックスを作成する対象となる別に対してインデックス、または動的インデックスがすでに作成されている場合の処理の例を示す。これは図4の表に対して次のSQL文が入力された場合の例である。

#### 【0059】

```
SELECT * FROM 会員表 WHERE 年齢=25 OR 年齢=30
```

図8ではすでに図7で作成されている動的インデックス、すなわち条件として年齢が25歳以上、30歳以下に対応する動的インデックスを用いて、更に条件として年齢が25歳、または30歳のデータだけが選択され、最終的な動的インデックスが作成される。初めから表4を対象として全件走査を行い、最終的な動的インデックスを作成するよりも、高速にインデックス作成を行うことができる。

## 【 0 0 6 0 】

このように本実施形態では、既に作成されている動的インデックスに新たな条件を追加して作成した動的インデックスを利用することによって、検索の高速化と、省資源が実現できる。

## 【 0 0 6 1 】

図 9 はすでに作成されている動的インデックスに対して更に探索条件として新たな条件を追加し、最終的な動的インデックスを作成する例である。すなわち図 7 において作成された動的インデックスに対して条件判断として年齢が 2 5 歳または 3 0 歳という判断を行い、更に探索条件として会員番号が A 3 0 0 0 以上 A 5 0 0 0 以下という判断を行って最終的な動的インデックスが作成される。この動的インデックスは次の S Q L 文の入力に対応して作成されるものである。

## 【 0 0 6 2 】

```
SELECT * FROM 会員表 WHERE (年齢 = 2 5 OR 年齢
= 3 0) AND 会員番号 ≥ ' A 3 0 0 0 ' AND 会員番号 ≤ ' A 5 0
0 0 '
```

図 1 0 は図 3 の管理辞書 4 1 の内容を示す。管理辞書 4 1 にはインデックスの名称、例えばインデックス①、インデックス②、動的インデックス①、・・・、インデックスが作成されている表の名称、例えば表①、インデックス①の作成条件、すなわち前述の条件や探索条件などが適用された列の名称、例えば列①、列②などが格納され、動的インデックスの場合には寿命管理のために列の名称に対応して動的インデックスの作成日時、作成条件（検索条件に相当）、サイズに加えて、アクセス回数、アクセス種別（参照、または更新・削除）の情報が保持されている。

## 【 0 0 6 3 】

また管理辞書の 4 1 の内部に格納されているインデックスの名称、および動的インデックスの名称は、それぞれ表領域 4 2 の内部のインデックス本体、作業領域 4 3 の内部の動的インデックス本体と対応づけられている。

## 【 0 0 6 4 】

管理辞書 4 1 の内部に格納されている動的インデックスのサイズは、後述するように動的インデックスが格納される作業領域の管理のためのものであり、作成日時は動的インデックスが作成された時点を示し、更にアクセス回数はそのインデックスが使用された場合に更新される。

#### 【 0 0 6 5 】

この管理辞書 4 1 へのデータの登録／削除は CREATE INDEX 文、DROP INDEX 文、および一意性制約定義／削除文など、インデックスを操作する DDL（データ デフィニション ランゲージ）文を用いて実行される。また動的インデックス作成装置 3 8 による動的インデックスの作成、寿命管理装置 4 0 の指示による動的インデックスの削除に対応して、管理辞書 4 1 への情報の登録／削除が行われる。

#### 【 0 0 6 6 】

なお図 3 ではインデックス管理装置 3 5 が、管理辞書 4 1 の内容を用いて、例えば動的インデックスの作成される領域の管理を行うものとしたが、その領域管理は動的インデックス作成装置 3 8 が行うこともできる。

#### 【 0 0 6 7 】

図 1 1 は動的インデックスの適用可能性判断処理のフローチャートである。同図において処理が開始されると、まずステップ S 2 1 でデータベースの検索条件に一致する動的インデックスが存在するか否かが判定され、存在しない場合には、ステップ S 2 2 で検索条件が範囲に含まれるインデックス、すなわち検索条件よりも広い条件のインデックスが存在するか否かが判定され、存在しない場合にはステップ S 2 3 で、すでに存在する複数のインデックスを組合わせることによって検索条件が含まれることになるような複数のインデックスが存在するか否かが判定され、そのような複数のインデックスが存在しない場合には、ステップ S 2 4 で動的インデックスの適用が不可能と判定されて、処理を終了する。

#### 【 0 0 6 8 】

これに対して、ステップ S 2 1 で検索条件に完全に一致する動的インデックスが存在すると判定された時、ステップ S 2 2 で検索条件よりも範囲の広い条件のインデックスが存在する時、ステップ S 2 3 で複数のインデックスの組合わせに

よって検索条件が含まれるようなインデックスの組合わせが存在する時には、ステップ S 2 5 で動的インデックスが適用可能であると判定されて、処理を終了する。なおステップ S 2 3 における複数のインデックスの組合わせについては後述する。

## 【 0 0 6 9 】

一般的に動的インデックスは不定形の検索に使用されるものであり、その意味では作成された時の検索条件でしか使用できない可能性が大きい。そこで動的インデックスの再利用は一般に困難であるが、本実施形態においては作成された動的インデックスの作成条件を記憶しておくことによって、動的インデックスの一部、または複数の動的インデックスの組合わせによって新たな検索条件に対応する検索が可能となる場合には、例えば動的インデックスの組合わせを利用して検索を行うことによって新たな動的インデックスを作成する必要がなくなり、動的インデックスを作成するための作業領域を小さくすることができ、また動的インデックスを新たに作成する処理が不必要となり、検索を高速に行うことが可能となる。

## 【 0 0 7 0 】

図 1 2 ～ 図 1 4 は動的インデックスの組合わせの説明図である。図 1 2 は作成済みの動的インデックスを示し、動的インデックスとして動的インデックス 1 から動的インデックス 4 が作成済みであり、それぞれ会員番号によって条件づけられていることが示されている。

## 【 0 0 7 1 】

図 1 3 は、図 1 2 の動的インデックス 1 から動的インデックス 4 までの会員番号の範囲と、新しい S Q L 文に対応する検索条件の範囲を示す。ここで新しい S Q L 文は次の文である。

## 【 0 0 7 2 】

```
SELECT * FROM 会員表 WHERE 会員番号 ≥ ' A 3 5
0 0 ' AND 会員番号 ≤ ' A 6 0 0 0 '
```

図 1 4 は動的インデックスの組合わせ処理のフローチャートである。同図において処理が開始されると、ステップ S 3 1 で検索条件を含むインデックスの組合

わせが存在するか否かが判定され、そのような組合わせが存在しない場合は、ステップ S 3 2 で組合わせの適用不可能とされて処理を終了する。

【 0 0 7 3 】

これに対して、前述の新しい S Q L 文に対応する条件、すなわち会員番号が A 3 5 0 0 以上で A 6 0 0 0 以下という検索条件を含むインデックスの組合わせとして、動的インデックス 1 と動的インデックス 2 との組み合わせ、または動的インデックス 1 と動的インデックス 4 との組合わせが存在すると判定され、ステップ S 3 3 でそのような組合わせを構成するインデックスのうちで検索条件が含まれる範囲の最も広いインデックスが検索される。これは範囲の広いインデックスを使用する方が、インデックスの適用切換えを行う回数が少なくなり、処理を高速化できるためである。

【 0 0 7 4 】

ここで、動的インデックス 1 は検索条件のうち会員番号としての A 4 5 0 0 から A 6 0 0 0 まで、動的インデックス 2 は A 3 5 0 0 から A 5 0 0 0 まで、動的インデックス 4 も A 3 5 0 0 から A 5 0 0 0 までと、いずれも同じ長さの範囲を持っているため、動的インデックス 1, 2, および 4 の 3 つが検索されたものとする。

【 0 0 7 5 】

次にステップ S 3 4 で、検索されたインデックスの中に同一範囲長を含むインデックスがあるか否かが判定される。ここでは動的インデックス 1, 2, および 4 がいずれも同一の範囲長を含むと判定され、続いてステップ S 3 5 で短いインデックスが使用される。これは同一範囲長を含むインデックスについては、インデックス全体が短い方が高速にアクセスできるためである。

【 0 0 7 6 】

ここでは動的インデックス 2 が、その全体の長さが最も短いインデックスとして使用されることが決定され、ステップ S 3 6 で検索条件のうち未適用の範囲があるか否かが判定される。

【 0 0 7 7 】

ここでは動的インデックス 2 のみが選択された状態であるため、検索条件のう



ちで会員番号 A 5 0 0 0 から A 6 0 0 0 までの範囲については未適用であり、ステップ S 3 3 に戻り、この条件の範囲を含むインデックスとして動的インデックス 1 が検索され、ステップ S 3 4 で同一範囲長を含むインデックスはないと判定され、ステップ S 3 6 で未適用の範囲もないと判定されて処理を終了する。

## 【 0 0 7 8 】

図 1 5 は動的インデックスの寿命管理処理のフローチャートである。本実施形態においては、新しい動的インデックスの作成にあたって、動的インデックスを作成するための作業領域に空きがあるかないかを判定し、空がない場合にはすでに作成された動的インデックスの消去を行うものとして、動的インデックスの寿命管理について説明するが、このような寿命管理は動的インデックスを作成するための作業領域に空きがあるか否かに無関係に、例えば随時行うこともできることは当然である。

## 【 0 0 7 9 】

図 1 5 において処理が開始されると、ステップ S 4 1 で動的インデックスを作成するための作業領域に空きがあるか否かが判定され、空きがある場合には、ステップ S 4 2 で新しい動的インデックスの作成が行われて、処理を終了する。

## 【 0 0 8 0 】

作業領域に空がない場合には、ステップ S 4 3 ですでに作成されている動的インデックスがアクセス回数によって降順にソートされ、ステップ S 4 4 でそのアクセス回数が最低の数の動的インデックスのグループが作成日時の順でソートされて、最も作成日時の古い動的インデックスが抽出される。

## 【 0 0 8 1 】

ステップ S 4 5 で、抽出された動的インデックスが更に更新頻度、すなわち更新回数のアクセス回数に対する比が大きい順にソートされ、ステップ S 4 6 で更新頻度の 1 番多いインデックスが消去される。これによってアクセス回数が最も少なく、作成日時が最も古く、更に更新頻度が最も多い動的インデックスから消去が行われることになる。

## 【 0 0 8 2 】

ステップ S 4 7 で、動的インデックスを作成するための作業領域がまだ不足し

ているか否かが判定され、まだ不足している場合にはステップ S 4 6 の処理に戻り、次に更新頻度の多いインデックスが消去される処理以降が繰返される。そしてステップ S 4 7 で動的インデックスを作成するための作業領域に空が出来たと判定されると、ステップ S 4 2 で新しい動的インデックスの作成が行われて、処理を終了する。

## 【 0 0 8 3 】

一般的にある検索条件に対応して作成された動的インデックスは、その再利用の可能性を期待してシステム内に保存されるが、動的インデックス作成用の作業領域の不足が発生するような場合には、作成済みのインデックスのうちで優先順位の低いものから削除することによって、頻繁に参照アクセスされるようなインデックスについては再利用が可能となり、頻繁に更新されるような動的インデックスは消去されやすくなり、動的インデックスの効率的な再利用が可能となる。

## 【 0 0 8 4 】

なお図 1 5 ではアクセス回数、作成日時、更新頻度の順番で動的インデックスの消去の優先順位づけを行っているが、この順位はシステムによってどのような順位に設定することも可能である。

## 【 0 0 8 5 】

以上説明したようなデータベースの検索方式は、当然一般的なコンピュータシステムによって実現することが可能である。図 1 6 はそのようなコンピュータシステムの構成ブロック図である。同図において、本発明のデータベース検索方式を実現するためのコンピュータ 8 1 は、本体 8 2 とメモリ 8 3 とによって構成されている。メモリ 8 3 はランダムアクセスメモリ (RAM)、ハードディスク、磁気ディスクなどの記憶装置であり、このようなメモリに本発明の特許請求の範囲第 1 ～ 5 項のプログラムや、図 5、図 6、図 1 1、図 1 4、図 1 5 のフローチャートで示したプログラムなどが格納され、そのプログラムが本体 8 2 によって実行されることによって、本発明のデータベース検索方式を実現することが可能となる。

## 【 0 0 8 6 】

このようなプログラムは、プログラム提供者側からネットワーク 8 4 を介して

コンピュータ 8 1 にロードされることも、また市販され、流通している可搬型記憶媒体 8 5 に格納され、そのような可搬型記憶媒体 8 5 がコンピュータ 8 1 にセットされることによって、本体 8 2 によって実行されることも可能である。

【 0 0 8 7 】

可搬型記憶媒体 8 5 としては C D - R O M 、 フロッピーディスク、光ディスク、光磁気ディスクなど様々な形式の記憶媒体を使用することができ、このような記憶媒体に前述のようなプログラムが格納されることによって、本発明におけるデータベース検索方式が実現可能となる。

【 0 0 8 8 】

(付記 1) データベースの検索を行う計算機によって使用されるプログラムにおいて、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手順と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、該検索条件に対応するインデックスを作成する手順と、

該作成されたインデックスを用いてデータベースの検索を行う手順とを計算機に実行させるためのプログラム。

(付記 2) データベースの検索を行う計算機によって使用されるプログラムにおいて、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手順と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みのインデックスのうちで、前記検索条件よりも広い条件を満足するインデックスが存在するか否かを判定する手順と、

該広い条件を満足するインデックスが存在する時、該インデックスを用いて該検索条件のみを満足するインデックスを作成する手順と、

該作成されたインデックスを用いてデータベースの検索を行う手順とを計算機に実行させるためのプログラム。

(付記 3) データベースの検索を行う計算機によって使用されるプログラムに

において、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手順と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みの複数のインデックスのうちで、組合わせることによって前記検索条件を満足する2つ以上のインデックスが存在するか否かを判定する手順と、

該2つ以上のインデックスが存在する時、該2つ以上のインデックスを組合わせて、該検索条件に対応するインデックスを作成する手順と、

該作成されたインデックスを用いてデータベースの検索を行う手順とを計算機に実行させるためのプログラム。

(付記4) 前記プログラムが、

前記作成されたインデックスに対するアクセス回数、作成日時、更新頻度のデータを管理する手順と、

該データの管理状況に対して、作成されたインデックスの消去を行う手順とを更に備えることを特徴とする付記1、2または3記載のプログラム。

(付記5) 前記プログラムが、

前記データベースへのアクセスがデータの更新、または削除のためのアクセスである時、該アクセス処理に適用可能な作成済みインデックスが存在するか否かを判定する手順と、

該インデックスが存在する時、該インデックスの存在によって該アクセス処理のアクセス性能が低下するか否かを判定する手順と、

該アクセス性能が低下する時、該アクセス処理の開始に先立って該インデックスを削除する手順とを更に備えることを特徴とする付記1、2、3または4記載のプログラム。

(付記6) データベースの検索方法において、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較し、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みの複数のインデックスのうちで、組合わせることによって前記検索条件を

満足する 2 つ以上のインデックスが存在するか否かを判定し、

該 2 つ以上のインデックスが存在する時、該 2 つ以上のインデックスを組合わせて、該検索条件に対応するインデックスを作成し、

該作成されたインデックスを用いてデータベースの検索を行うことを特徴とするデータベース検索方法。

(付記 7) データベースの検索を行う計算機によって使用される記憶媒体において、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全体検索を行う場合とのコストを比較するステップと、

該コスト比較の結果、全体検索を行う場合の方がコストが大きい時、すでに作成済みの複数のインデックスのうちで、組合わせることによって前記検索条件を満足する 2 つ以上のインデックスが存在するか否かを判定するステップと、

該 2 つ以上のインデックスが存在する時、該 2 つ以上のインデックスを組合わせて、該検索条件に対応するインデックスを作成するステップと、

該作成されたインデックスを用いてデータベースの検索を行うステップとを計算機に実行させるためのプログラムを格納した計算機読出し可能可搬型記憶媒体

(付記 8) データベースの検索を行うデータベース検索装置において、

検索条件に対応するインデックスを作成した後に検索を行う場合と、全件検索を行う場合とのコストを比較する手段と、

該コスト比較の結果、全件検索を行う場合の方がコストが大きい時、すでに作成済みの複数のインデックスのうちで、組合わせることによって前記検索条件を満足する 2 つ以上のインデックスが存在するか否かを判定する手段と、

該 2 つ以上のインデックスが存在する時、該 2 つ以上のインデックスを組合わせて、該検索条件に対応するインデックスを作成する手段と、

該作成されたインデックスを用いてデータベースの検索を行う手段とを備えることを特徴とするデータベース検索装置。

【 0 0 8 9 】

【発明の効果】 以上詳細に説明したように本発明によれば、データベース

の検索に必要なインデックスがない場合、またはインデックスがあってもそのままでは検索のために使用できない場合に、検索に必要なインデックスを動的に作成して検索を行うことができ、データベースの検索性能の向上に寄与するところ  
が大きい。

【 0 0 9 0 】

一般に特定の条件に対して作成されたインデックスは、その条件に一致する検索条件に対してしか適用できないが、本発明によれば特定の条件のうちの一部の条件だけに適用することや、複数のインデックスを組合わせて適用することなどが可能となり、インデックス作成用作業領域の削減、インデックス作成時間の短縮、データベース検索処理時間の短縮を実現することが可能となる。更に動的インデックスの寿命管理を行うことにより、動的インデックスの有効な再利用が可能となり、また大量のデータ更新時には動的インデックスを削除することによって、アクセス性能の劣化を防ぐことができる。

【図面の簡単な説明】

【図 1】

本発明の機能ブロック図である。

【図 2】

本発明のデータベース検索システムの基本構成ブロック図である。

【図 3】

本発明の S Q L 処理システムの構成を示すブロック図である。

【図 4】

データベースの表の例である。

【図 5】

データベースに対する参照アクセス処理のフローチャートである。

【図 6】

データベースに対する更新・削除アクセス処理のフローチャートである。

【図 7】

動的インデックスの作成例（その 1）を示す図である。

【図 8】

動的インデックスの作成例（その 2）を示す図である。

【図 9】

動的インデックスの作成例（その 3）を示す図である。

【図 1 0】

インデックス管理装置に対応する管理辞書の格納内容の例を示す図である。

【図 1 1】

動的インデックス適用可能性判定処理のフローチャートである。

【図 1 2】

動的インデックスの例を示す図である。

【図 1 3】

図 1 2 の動的インデックスの範囲を示す図である。

【図 1 4】

動的インデックスの組合わせ処理のフローチャートである。

【図 1 5】

動的インデックスの寿命管理処理のフローチャートである。

【図 1 6】

本発明を実現するためのプログラムのコンピュータへのローディングを説明する図である。

【符号の説明】

- 1 0     データベース検索装置
- 1 1     アプリケーションプログラム
- 1 2     構文解析部
- 1 3     アクセス処理最適化部
- 1 4     アクセス処理部
- 1 5     インデックス管理部
- 1 6     表
- 1 7     インデックス
- 1 8     インデックス作成／消去部
- 1 9     動的インデックス

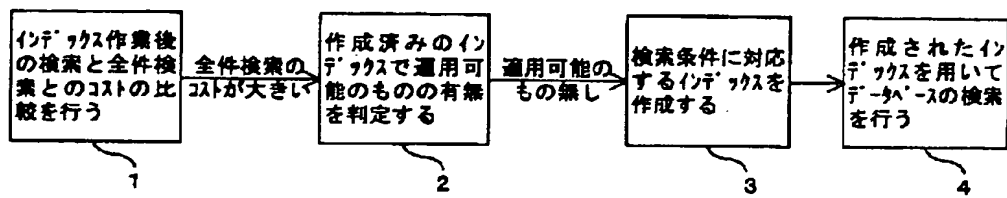
- 2 0 寿命管理部
- 3 0 S Q L 处理装置
- 4 1 管理辞書
- 4 2 表領域
- 4 3 作業領域



【書類名】 図面

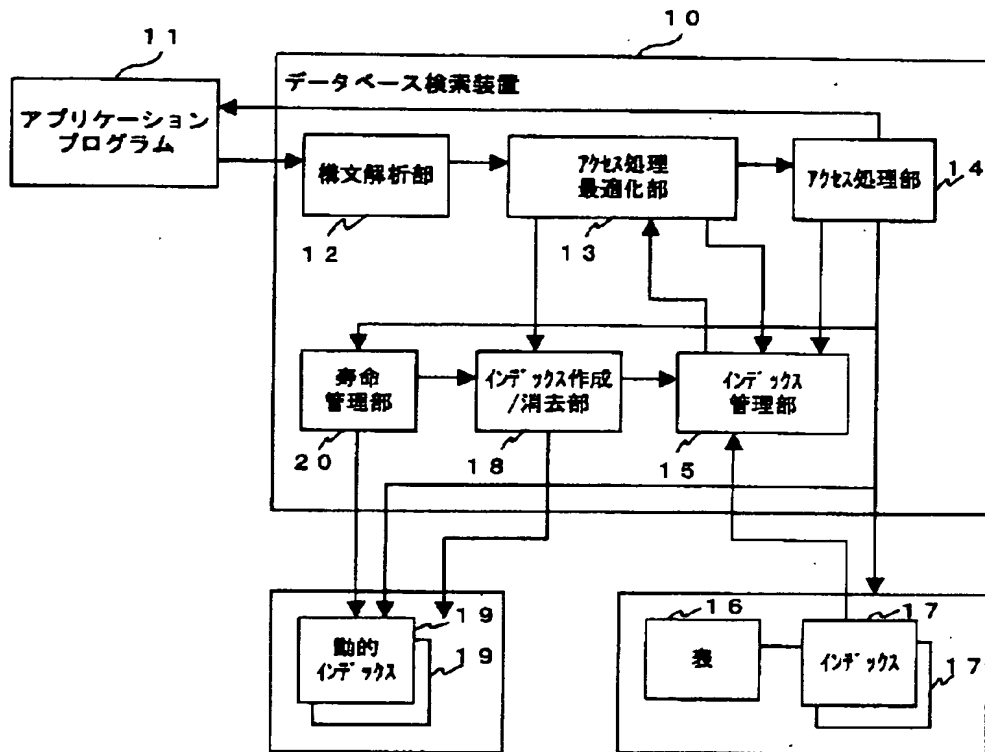
【図 1】

### 本発明の機能ブロック図



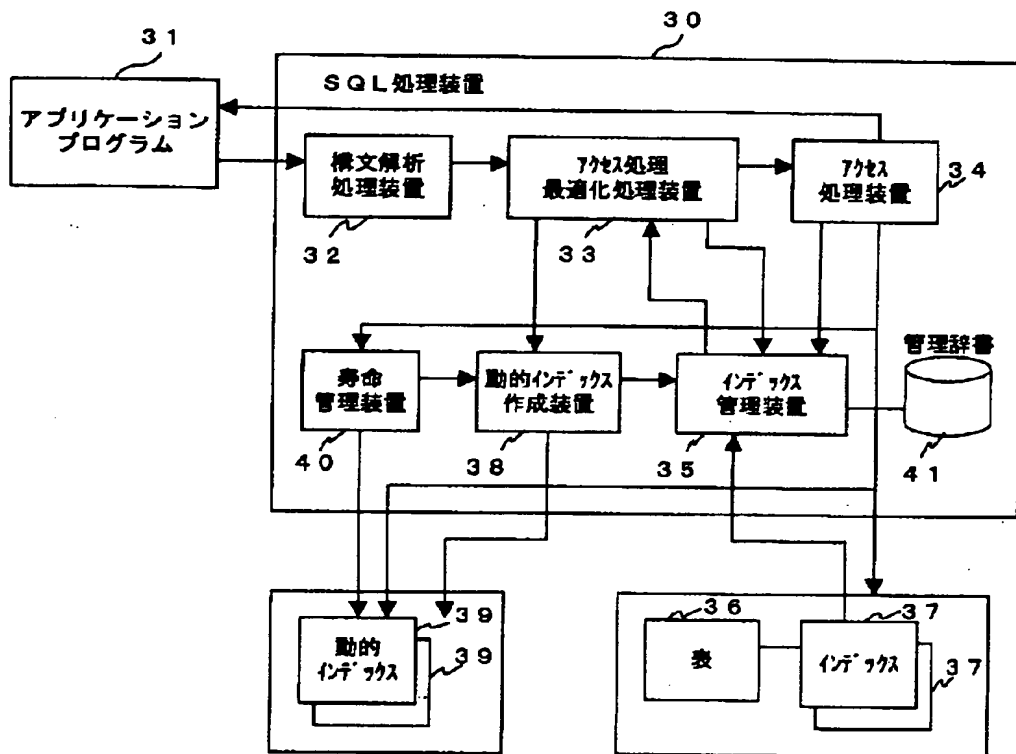
【図 2】

本発明のデータベース検索システムの基本構成ブロック図



【図 3】

本発明のSQL処理システムの構成を示すブロック図



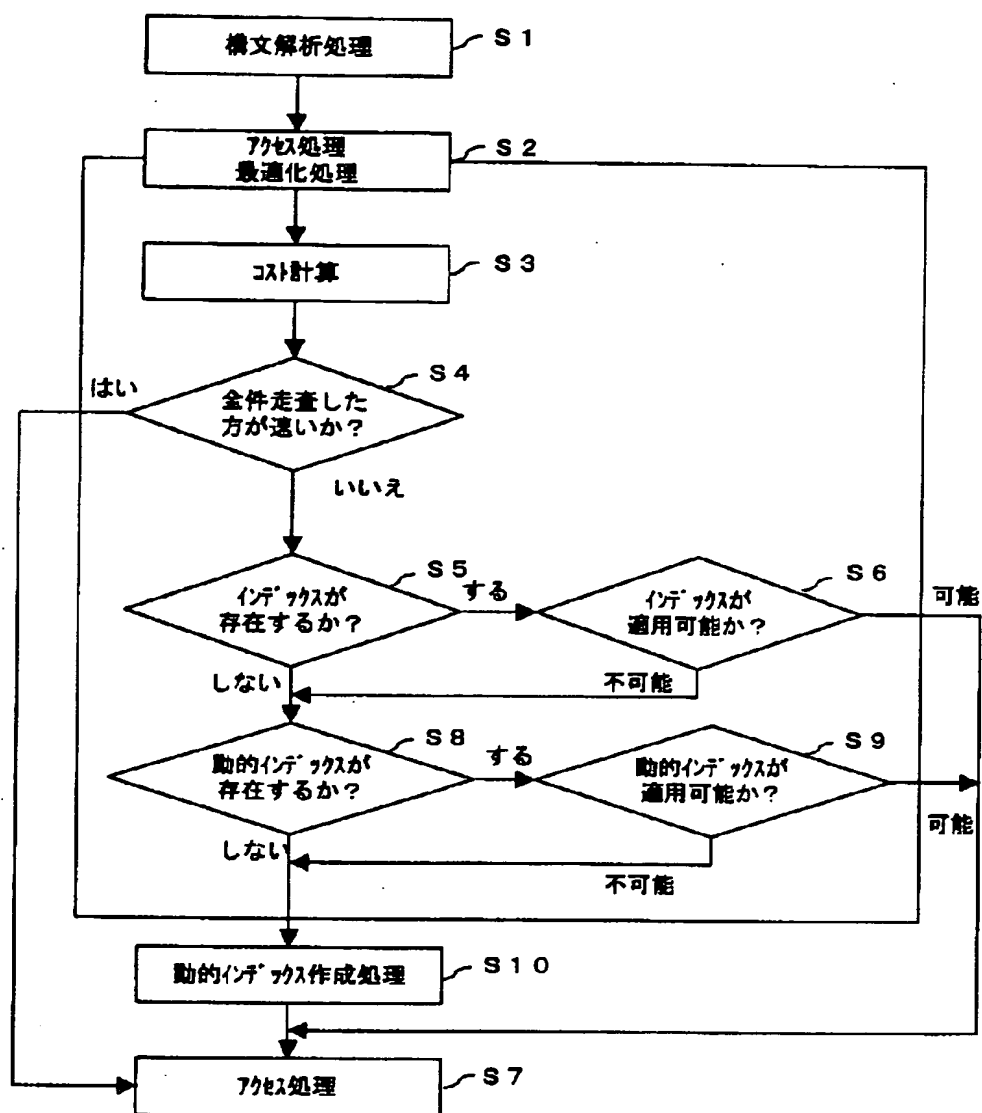
【図 4】

# データベースの表の例

会員番号	年齢	氏名
A0001	31	八木田
A0002	35	森田
A0003	28	吉山
A0004	31	田村
~	~	~
A5230	29	荒川
~	~	~
A9998	32	岸本
A9999	30	福留

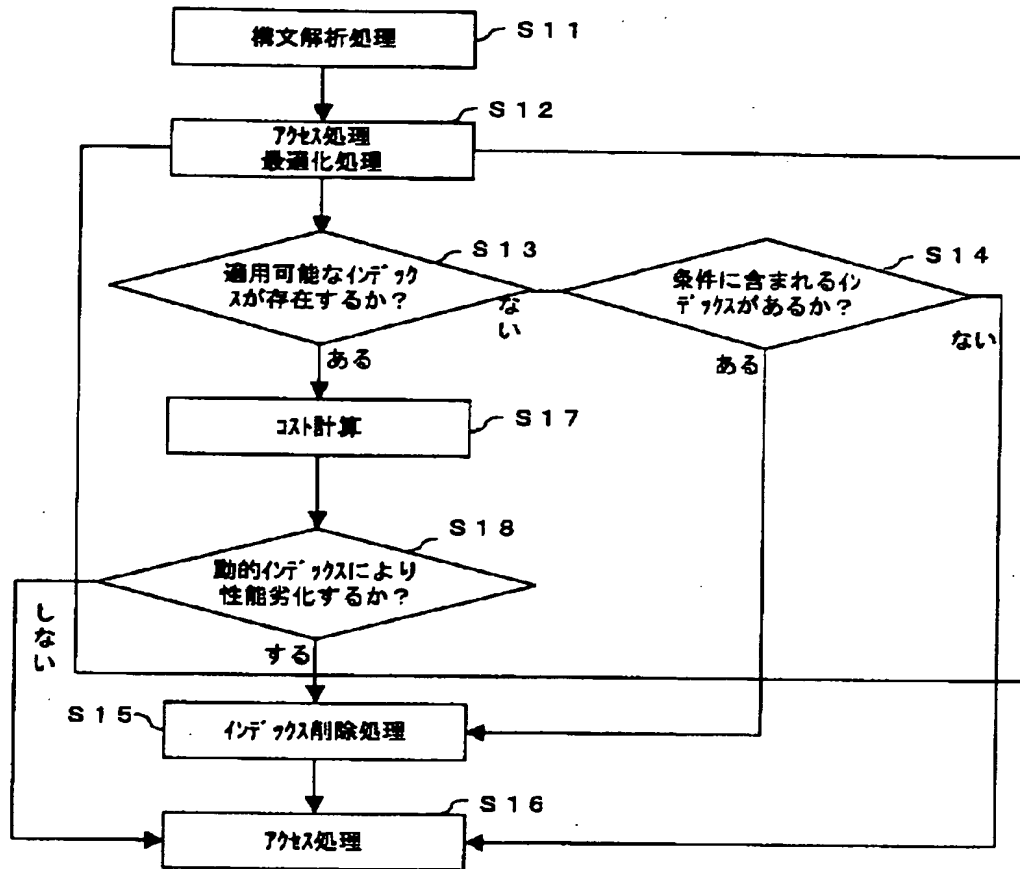
【図 5】

データベースに対する参照アクセス処理のフローチャート



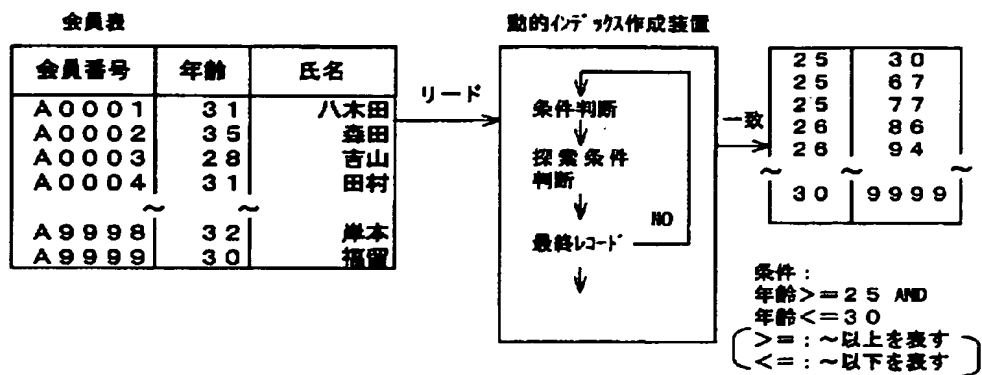
【図6】

データベースに対する更新・削除  
アクセス処理のフローチャート



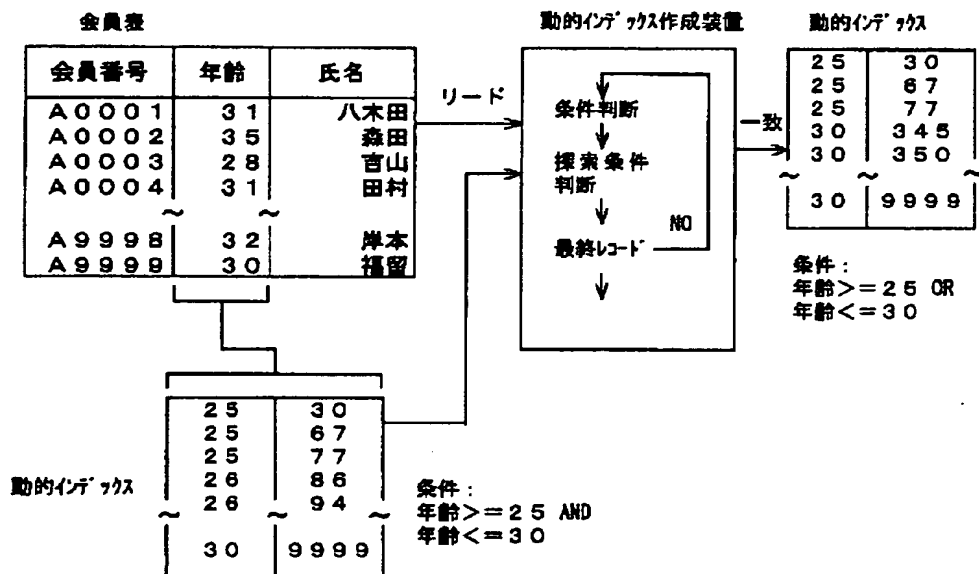
【図 7】

動的インデックスの作成例（その 1）を示す図



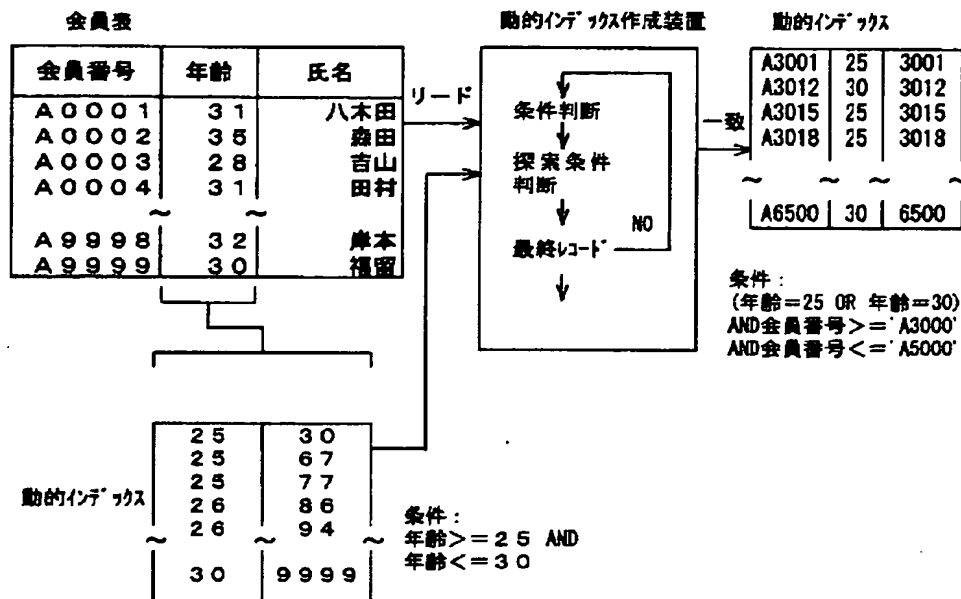
【図 8】

動的インデックスの作成例（その 2）を示す図



【図9】

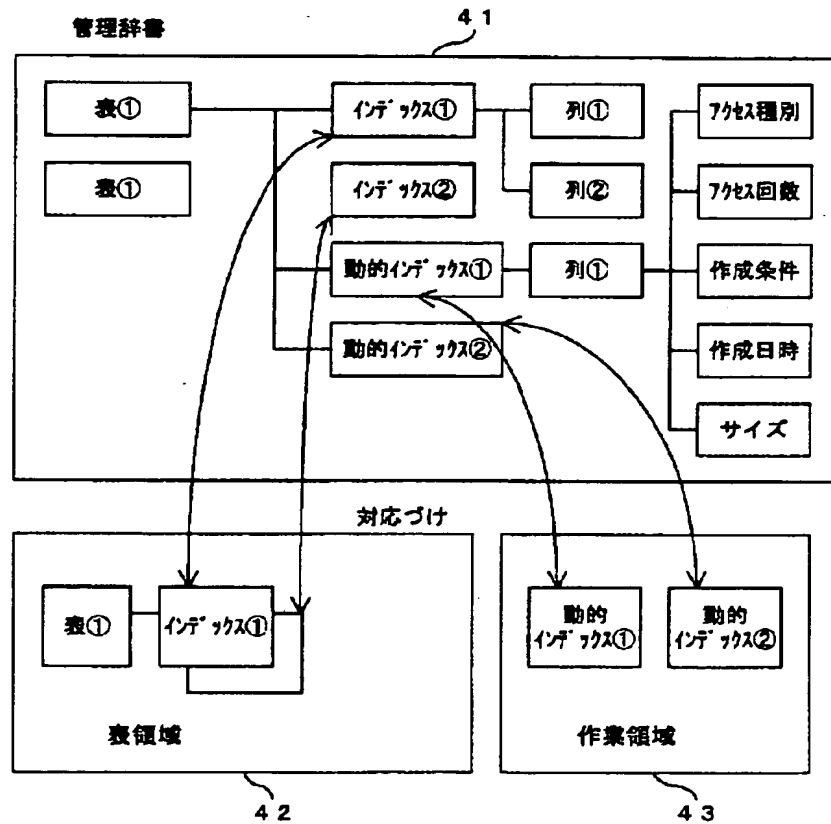
動的インデックスの作成例（その3）を示す図





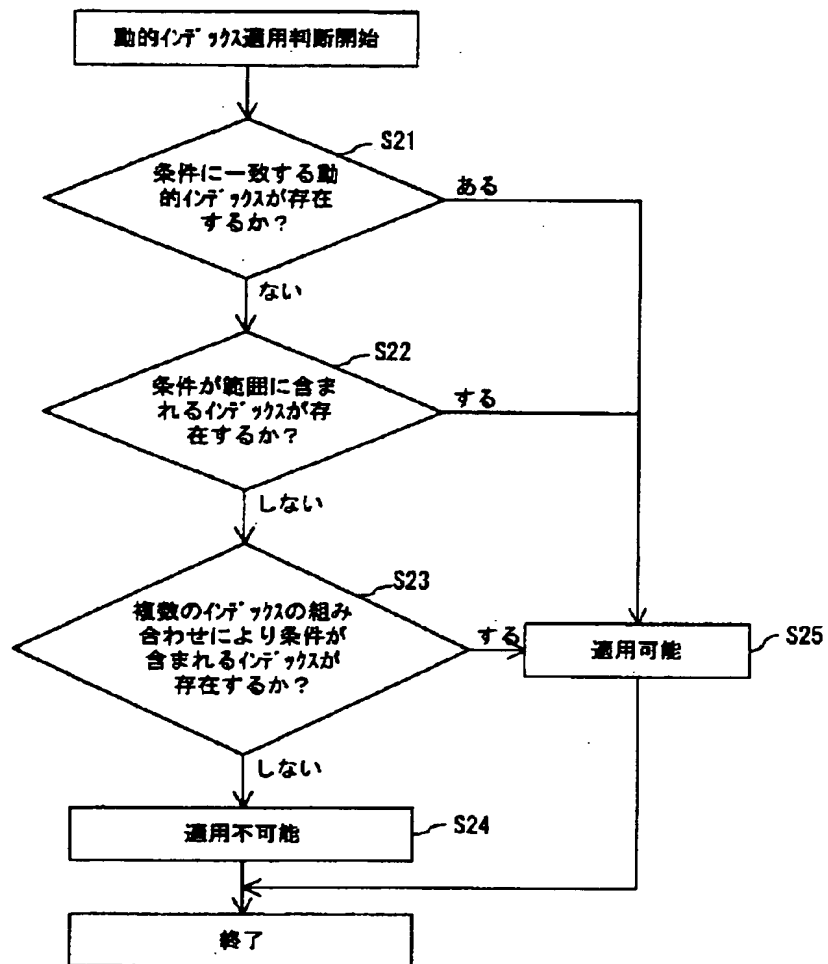
【図10】

インデックス管理装置に対応する管理辞書の格納内容の例を示す図



【図 11】

動的インデックス適用可能性判定処理の  
フローチャート



【図 1 2】

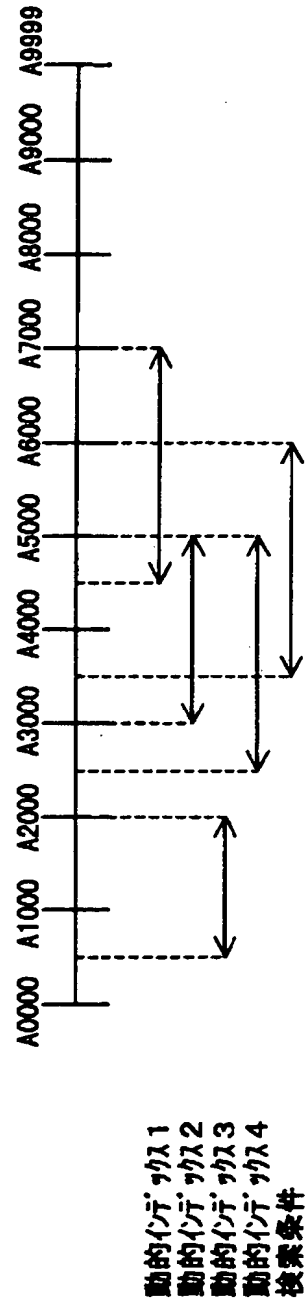
動的インデックスの例を示す図

	作成済み動的インデックス
動的インデックス 1	会員番号 >= 'A4500' AND 会員番号 <= 'A7000'
動的インデックス 2	会員番号 >= 'A3000' AND 会員番号 <= 'A5000'
動的インデックス 3	会員番号 >= 'A0500' AND 会員番号 <= 'A2000'
動的インデックス 4	会員番号 >= 'A2500' AND 会員番号 <= 'A5000'

>= : ~以上を表す  
<= : ~以下を表す

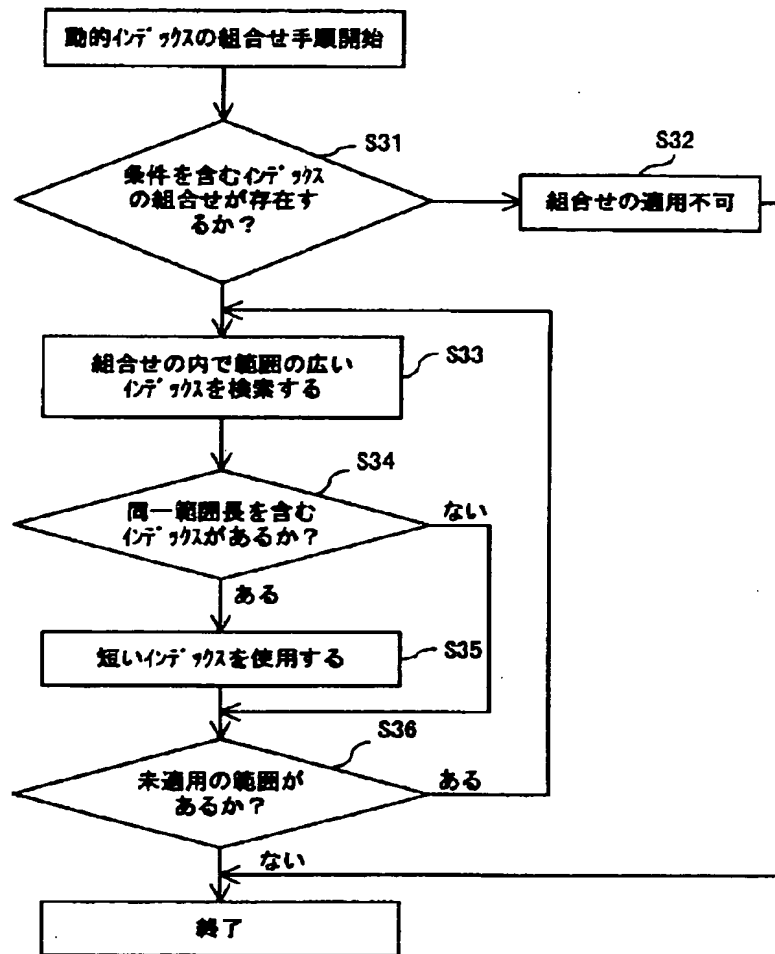
【図 1 3】

図 1 2 の動的インデックスの範囲を示す図



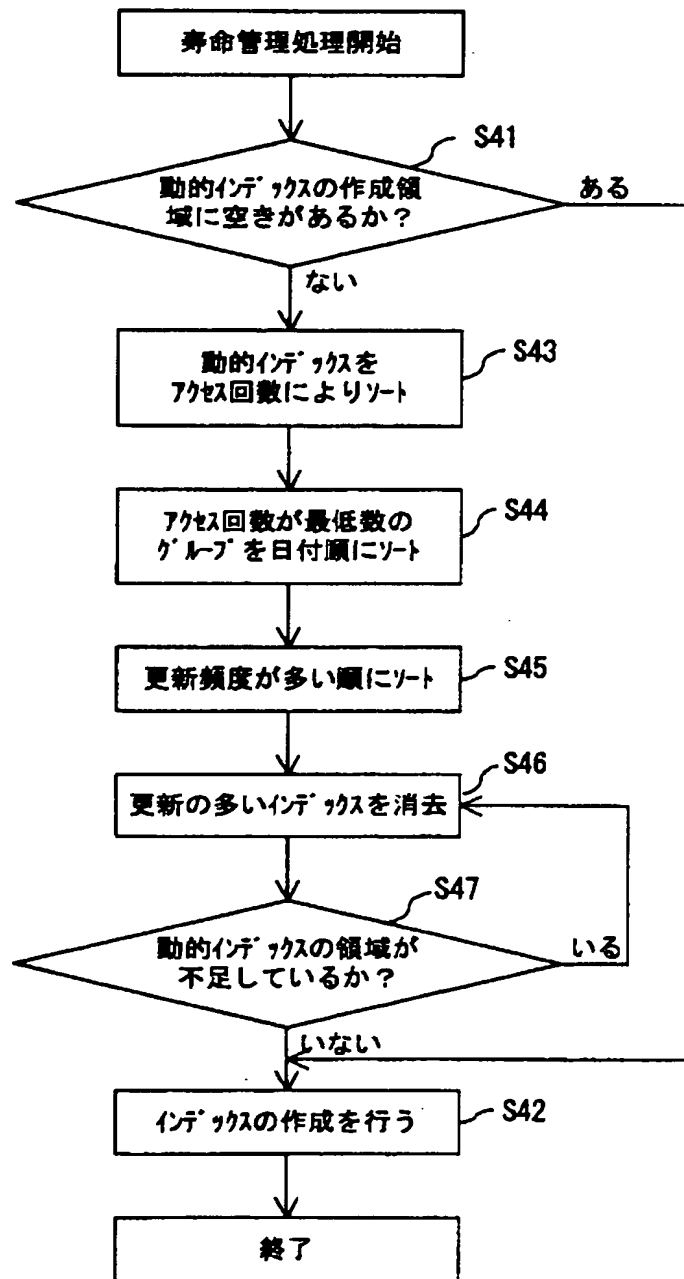
【図 14】

動的インデックスの組合せ処理のフローチャート



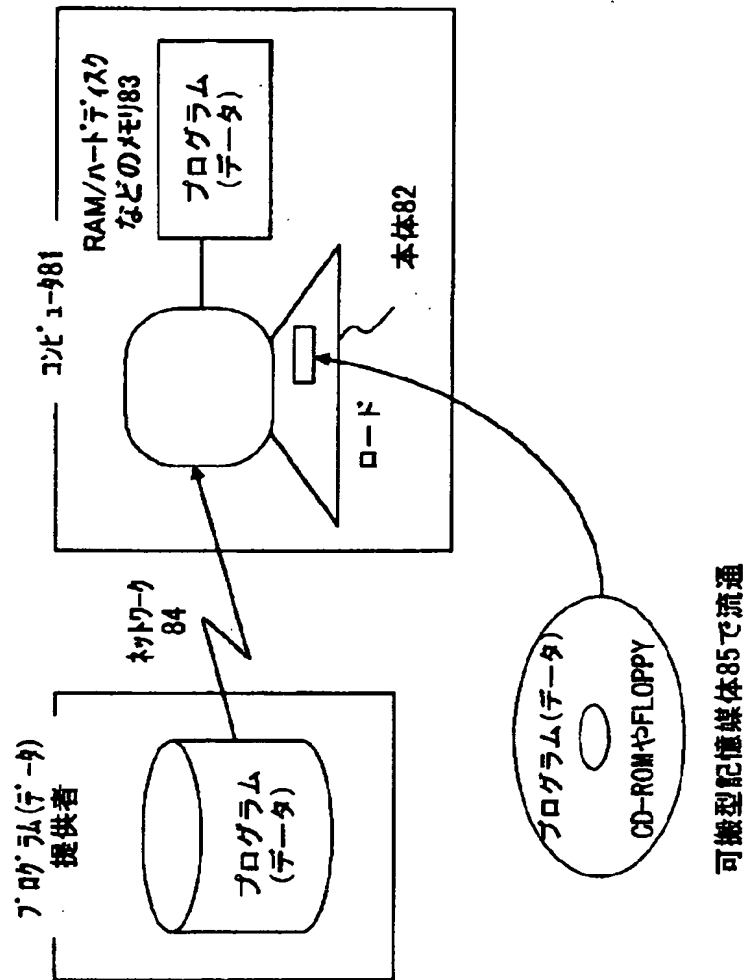
【図15】

# 動的インデックスの寿命管理処理の フローチャート



【図16】

本発明を実現するためのプログラムのコンピュータ  
へのローディングを説明する図



【書類名】 要約書

【要約】

【課題】 インデックスが存在しないか、または存在しても使用できない場合のデータベース検索を高速化する。

【解決手段】 検索条件に対応するインデックスの作成後に検索を行う場合と全件検索を行う場合とのコストを比較し、全件検索の方がコストが大きい時、既に作成済みのインデックスのうちで適用可能なインデックスの存在の有無を判定し、適用可能なインデックスが存在しない時、検索条件に対応するインデックスを作成し、作成されたインデックスを用いてデータベースを検索する検索方法およびこの方法を実行するための記憶媒体を用いる。

【選択図】 図 1



出 願 人 履 歴 情 報

識別番号 [000005223]

1. 変更年月日	1996年 3月26日
[変更理由]	住所変更
住 所	神奈川県川崎市中原区上小田中4丁目1番1号
氏 名	富士通株式会社

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**